Claim Amendments

Claim 1 (previously presented): A switch for transferring data comprising:

at least one master unit;

a plurality of slave units;

a bus through which the master unit communicates with the slave units; and

a memory in communication with the master unit having a software program that causes the master unit to automatically recover and restart when a slave unit fails which has caused the master unit to fail and to avoid further accessing the failed slave unit.

Claim 2 (previously presented): A switch as described in Claim 1 including persistent storage that survives across abnormal termination of the master unit.

Claim 3 (previously presented): A switch as described in Claim 2 including a mechanism for detecting failures of the slave units and thereupon causes the master unit to abnormally terminate.

Claim 4 (previously presented): A switch as described in Claim 3 wherein the software program causes the master unit to automatically recover when the detecting mechanism causes the master unit to abnormally terminate.

Claim 5 (original): A switch as described in Claim 4 wherein the detecting mechanism includes a hardware watchdog device.

Claim 6 (currently amended): A method for transferring data comprising the steps of:

attempting to access a failed slave unit of a plurality of slave units of a switch by a master unit of the switch with a signal through a bus through which the master unit and the failed slave unit communicate; and

automatically recovering and restarting the master unit which has failed because the failed slave unit failed and caused the master unit to fail with a software program in the switch that directs the master unit to avoid further accessing the failed slave unit of the plurality of slaver units.

Claim 7 (original): A method as described in Claim 6 wherein the recovering step includes the step of obtaining status information about the slave units from persistent storage.

Claim 8 (currently amended): A software program that is stored on computer readable medium whose contents causes a processor to perform the steps of:

determining a master unit abnormally terminated when the master unit attempted to access a first slave unit which caused the master unit to fail;

restarting the abnormally terminated master unit;

identifying the first slave unit of a plurality of slave units of a switch has failed when the first slave unit is attempted to be accessed by the master unit of the switch; and

preventing the master unit from attempting to access the failed first slave unit.

Claim 9 (canceled)

Claim 10 (previously presented): A program as described in Claim 8 including the step of changing information in persistent storage associated with the first slave unit from identified as failed to identified as good if the master unit does not terminate abnormally after the master unit attempts to contact the slave unit.

Claim 11 (original): A program as described in Claim 10 including the step of setting a variable slot chosen from amongst a plurality of slots of the switch not marked as potentially bad.

Claim 12 (original): A program as described in Claim 11 including the step of determining whether the first slave unit is physically present in a first slot of the plurality of slots.

Claim 13 (original): A program as described in Claim 12 including the step of determining the first slot is marked to be skipped.

Claim 14 (original): A program as described in Claim 13 including the step of marking the variable slot as potentially bad if it is not marked potentially bad.

Claim 15 (original): A program as described in Claim 14 including the step of reporting the variable slot as containing broken hardware and preventing the master unit from attempting to access the variable slot if the variable slot is marked to be skipped.

Claim 16 (original): A program as described in Claim 15 including the step of attempting to access hardware present in the variable slot if the variable slot is marked potentially bad.

Claim 17 (previously presented): A program as described in Claim 16 including the step of marking the variable slot as good if the master unit did not abnormally terminate when the master unit accessed the first slave unit.

Claim 18 (original): A program as described in Claim 17 including the step of enabling normal operations on hardware present in the variable slot if the variable slot is marked as good.

Claim 19 (original): A program as described in Claim 18 including the step of setting the variable slot to a next slot of the plurality of slots.